

Grafos Que Utilizamos Para Testear

Al momento de realizar la experimentación con respecto a los grafos randoms, surgió la idea de utilizar en cada instancia los mismos grafos random. Cada test se basaba en realizar para cada tamaño N , 40 grafos distintos, realizar el experimento en cada uno de ellos, y luego promediar los resultados.

La primera idea de como hacer esto fue el de almacenar todos los grafos en un txt y luego leerlos en cada experimento. Sin embargo al momento de implementar esto surgieron dos problemas

- La complejidad de adaptar todas las funciones para que lean el archivo donde tenemos almacenados los grafos, y traducir lo que se lea a las estructuras de representación que utilizamos.
- El tamaño del archivo resultante. Sin ir más lejos tan pronto creamos un archivo con grafos de hasta n igual a 300 con 40 ejemplos de cada instancia, nos quedo un archivo de 250 MB, que noteniamos forma de pasarnoslo, y siquiera de abrirlo.

Como solución a los problemas planteados, surgió que generemos los grafos en código durante la ejecución de los test.

Ahora para que en cada ejecución de los experimentos se generen los mismos 40 grafos sobre cada n . Lo que hicimos fue agregarle como parámetro a la funcion *GenerarGrafoRandom* la semilla de la cual queremos que comience a devolver número random.

Luego para no sacar el factor random con el que se generan, implementamos que la semilla que se pase al generador, sea justamente un numero random también. Y por último para asegurarnos que la semilla para cada grafo sobre cada n sea la misma en cada ejecución del algoritmo, comenzamos siempre la ejecución de nuestros experimentos plantando la semilla 60. Numero elegido arbitrariamente.

Para finalizar y mantener la congruencia en todos los experimentos surgió como convención que se utilicen siempre las mismas iteraciones para cada experimento, generando que siempre que se llegue al momento de generar los grafos de una instancia n . Sea el mismo numero de veces que se llamo a la función *rand()* con el fin de pasar la semilla al generador. En lineas generales todos nuestros experimentos siguen el siguiente pseudo código:

Algorithm 1 Algoritmo Backtracking sin Poda

```
function ExperimentoX
  srand(60)
  for  $i$  in [1..MaximaCantidadDeNodosQueQuieroExperimentar] do
    for  $j$  in [0..40) do
      Graph  $g$  = GenerarGrafoRandom( $i$ ,rand())
      .
      .
      .
      RESTO DEL EXPERIMENTO
      .
      .
    end for
  end for
end function
```
